

Real Time Data (RTD) function in Office XP

Created: 7/3/2001

Revised: 7/20/2001

Introduction

Microsoft Excel XP (also known as Excel 2002 or version 10) provides a new worksheet function called RTD that allows the retrieval of data in real time. This function is specifically designed to access real-time data from market data vendors, such as Reuters and Bloomberg. In previous versions of Excel, this was accomplished by using DDE (Dynamic Data Exchange) or through custom developed worksheet functions, such as BLP (Bloomberg) or RTGet (Reuters). While these methods have been used successfully, they have limitations and are not always as robust as one would like. It seems likely that Microsoft intends to replace DDE with RTD as the preferred method of bringing real-time data into Excel. Microsoft claims that RTD is superior to DDE in robustness, reliability, and convenience. We will have to wait for the implementation of real-world RTD servers before we can pass judgment on this claim. This paper contains a brief discussion of how RTD works and how it compares to other real-time data mechanisms for Excel.

What About DDE?

Although DDE has been used to successfully implement many real-time solutions, including those involving large amounts of frequently updating financial data, it does have significant limitations. When the number of DDE formulas in a sheet becomes very large, performance can be greatly reduced. DDE formulas also lack a cell referencing capability, a shortcoming that greatly hampers ease of use. In addition, the cryptic syntax of the DDE formula makes implementation difficult for the typical Excel user. Many tools have been developed and distributed by market data vendors and third party software developers to hide these shortcomings from the user. Developers have also created worksheet functions to bring real-time data into Excel with varying degrees of success. Here again, performance and reliability have not been optimal.

Promise of RTD

Is RTD the tool that developers of Excel real-time solutions have been clamoring for? The RTD function updates in real time and provides cell-referencing capabilities. It takes care of the overhead of routing data returned from the server to the correct cell in the worksheet containing the RTD function. In addition, Microsoft ~~also~~ claims that it has a higher throughput than DDE and will not drop updates when Excel is busy. However, much will depend upon the development and implementation of the RTD server – a component that will most likely be developed by each market data vendor individually. Just like DDE servers, some vendors do a better job than others.

The claim of greater efficiency is made based on the *hybrid push pull mechanism* on which the architecture of RTD is built. The RTD server signals Excel when data is available – the push. Excel retrieves the data when it is ready – the pull. Microsoft claims that all previous real-time solutions for Excel were either push only or pull only. While RTD does appear to be an improvement over DDE, only time will tell. Market data vendors must develop RTD server implementations and only then will we be able to thoroughly test performance and reliability.

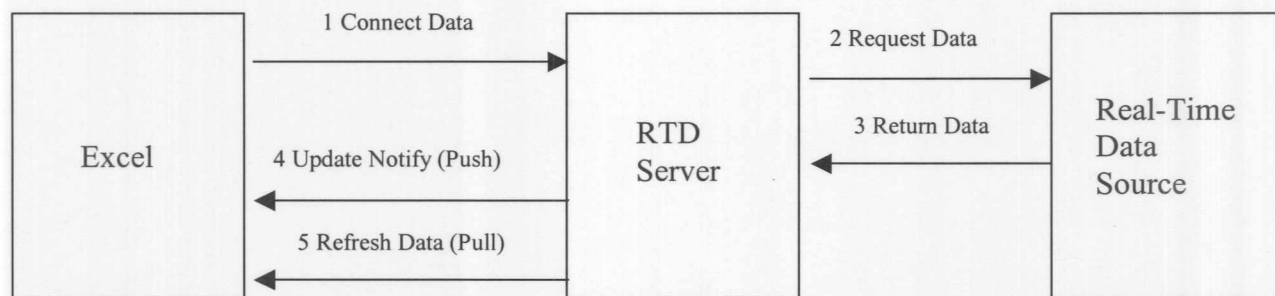
Behind The Scenes

In order to bring data into a spreadsheet using the RTD function, you must have an RTD server to supply the data. An RTD server is a Component Object Module (COM) Automation server. In other words, it is a DLL or EXE that receives requests for data from Excel, passes the request on to some real-time data source, retrieves data from the source, and passes the data back to Excel.

The server must implement the `IrtdServer` interface that is new in Excel XP. This is simply a class module that provides code for the following 6 methods:

<code>ConnectData</code>	Called by Excel to request a new topic from the server when an RTD functions is placed in a worksheet.
<code>DisconnectData</code>	Called when Excel no longer needs a topic – when an RTD function is deleted or Excel closes.
<code>Heartbeat</code>	Called by Excel to find out if the RTD server is still running. If Excel does not receive the proper response, then it displays a message box asking if it should attempt to reconnect to the server.
<code>RefreshData</code>	Called by Excel to get new values for RTD functions from the server after an update notify message is received from the server.
<code>ServerStart</code>	Called by Excel the first time that a topic is requested.
<code>ServerTerminate</code>	Called by Excel when no more topics are required – when all RTD functions are deleted or Excel is closed.

The author of an RTD server must supply code for these methods, and will determine what kind of data the RTD server will provide. Excel will automatically call these methods to return data to the cells containing RTD functions. No code is required in Excel to accomplish this. The diagram below shows the order of the events that takes place when an RTD function is placed in a worksheet.



First, Excel makes a call to the `ConnectData` method of the RTD server. Excel passes the parameters of the RTD function to this method, requesting that the server do something with the information. The developer can design the RTD server to return whatever data is relevant to the needs of a particular application. In the case of real-time market data, the request might be for a stock symbol and a field, such as the last price of IBM. The RTD server would then make a request to the real-time market data source for the last price of IBM. The request to the data source would ideally be by subscription. The real time data source would send data every time IBM's price changes. The RTD server could be designed to store every price until Excel pulls all of the data, or it could save only the latest data and throw away the rest.

When the RTD server has new data for Excel, it calls the `UpdateNotify` method of the `IrtdUpdateEvent` interface. This is another new interface in Excel XP. You do not need to write any code for this interface; Excel instantiates the object when it sees an RTD function. The `Notify` event simply lets Excel know that the data it requested earlier is available. If Excel is busy, because a message box is being displayed or a user is typing into a cell, it will wait until time is available and then pull the data from the RTD server by calling the `RefreshData` method of the `IrtdServer` interface.

The actual data to be returned is determined by the RTD server. Data for any number of RTD functions can be returned at one time. There does not need to be a one-to-one correspondence between the number data requests and the number of data returns. Excel will accept any number of RTD return values and automatically update all of the appropriate cells. The structure of the data returned to Excel is a two-column array. The first column contains the unique identifiers which Microsoft calls **Topic ID** that enable Excel to locate the cells in which to put the data. The second column contains the data corresponding to the applicable Topic ID. The Topic ID is automatically generated by Excel and passed to the RTD server when Excel makes a `ConnectData` request. The server must remember this ID and pass it back with the data corresponding to the applicable Excel request.

Very little code is needed to implement an RTD server. It consists mainly of a class module containing the events that Excel will call to request data from the server. The other code of significance in a market data server is the component that supplies the real-time data.

How They Compare

The table below is a comparison of features between RTD, DDE, and three sample data functions, Reuters RtGet, Reuters RtUpdate function, and the Willow Table function.

Description	RTD	DDE	RtGet	RtUpdate	Wtable	Comments
Allows updates with macro security set to High	No	Yes	No	No	Yes	Office XP macro security defaults to High when installed. (See note 1)
Allows updates with spreadsheet Calculation set to Manual	No	No	No	No	No	
Allows updates with Remote References turned off	Yes	No	Yes	Yes	No	
Data link listed under Edit Links .	No	Yes	No	No	Yes	RTD is not a link. (See note 2)
Generates links startup message when workbook is opened	No	Yes	No	No	Yes	
Allows cell referencing	Yes	No	Yes	Yes	Yes	
Allows formulas in function arguments	Yes	No	Yes	Yes	No	DDE has no formal arguments; it is a single string and cannot contain any calculations.
Can return multiple values	Yes	Yes	No	Yes	Yes	RtGet returns only a single value. (See note 3)
Truly Real Time	Yes	Yes	No	No	Yes	
Can specify time interval for updates	Yes	Yes/No	Yes	Yes	Yes/No	Reuter and Bloomberg DDE yes, Bridge DDE no. (see notes 4 & 5)

Notes

1. When Excel macro security is set to **High**, RTD functions return #N/A without any other indication as to why there is no data. The default setting for macro security when installing Office XP is **High**. After Excel is installed, RTD functions return #N/A and it is not obvious that the macro security needs to be reduced in order to receive valid data. Also, with macro security set to high, the Adfin add-in does not load so RtGet and RtUpdate are not available.
2. An RTD *link* to a server is not the same as a DDE *link*. RTD links are not listed in the Excel links list. You cannot view or change the status of an RTD function through the **Edit Links** dialog box as can be done with DDE formulas.
3. The RTD function cannot return an array of values as can be done in DDE. As a result, if an RTD function is used in a worksheet array formula, all of the cells return #VALUE. An RTD server can be written to return multiple values by combining the values into one delimited string. A VBA wrapper function can then be used to parse the string into an array that can be put into the worksheet. If the server uses a comma for as a column separator, semi-colon for row separator and encloses the entire return string in curly brackets, then the string can be easily parsed by using the Application.Evaluate method of Excel.
4. The RTD object provides a mechanism for controlling the frequency of updates in Excel. It has a ThrottleInterval property that allows the frequency to be specified in milliseconds. This interval has a global affect; it is applied to all RTD functions in Excel. (Application.RTD.ThrottleInterval = 2000 is the default). Setting the interval to -1 stops updates of all RTD functions in Excel. A setting of 0 tells Excel to accept updates whenever they are available. The throttle interval value persists for future Excel sessions. If the interval is changed, Excel remembers the new value the next time it is opened. The throttle interval time is stored in the Windows registry.

5. PDD (Reuters) and Bloomberg DDE have settings that enable a user to set the update interval for real-time data on their workstation. The Willow table function uses DDE formulas so its ability to specify update intervals depend on the DDE flavor being used.

Points of Interest

1. The RTD function behaves differently than other worksheet functions. Its function value is updated whenever new data becomes available from the RTD server and the workbook is able to accept it (when Excel is not busy and the RTD throttle interval has elapsed). Other functions are updated only when the worksheet is recalculated.
2. An RTD server can reside on your local machine or on a different machine. The machine on which it runs is specified in the second argument of the RTD function. Leave this blank if running the RTD server on your local machine. When on a different machine, the RTD server communicates with Excel through DCOM.
3. According to Microsoft, the performance of RTD in Excel with a server running on a Pentium 3 at 500 MHz with 128 MB of RAM is as follows:
20,000 unique topics updated 3 times per second or
1 topic updated 200 times per second
The frequency of updates for one topic is limited by the maximum rate at which Excel can receive Windows Messages.
4. Performance is also a function of how often the RTD server can supply updates to Excel. This is dependent on the design of the server. The server can call the UpdateNotify method as often as it wants. This tells Excel data is available. Excel will only request data when it is ready.
5. The RTD throttle interval can be useful in preventing Excel from becoming too busy for the user to do anything else. If the server provides updates too often, Excel may become stuck in a state where it is spending most of its time updating RTD and calculating, making it almost impossible for the user to work. In this situation, the throttle interval can be adjusted by the user to restrict how often Excel will request RTD updates.
6. The use of a wrapper function can make the RTD function easier to use. It hides the details of the RTD function parameters for server name and ID. The RealTimeServerProgID tells Excel where to find the RTD server code that implements the IrtDServer interface. It is a string consisting of the name of the DLL or EXE containing the RTD server, a dot separator, and the name of the class module in which the IrtDServer methods have been defined (ex. MyDLLName.MyClassModuleName)

RTD function:

=RTD(RealTimeServerProgID,MachineNameWhereServerRuns,Topic1,[Topic2],...)

Wrapper function:

=GetMyData(MSFT, LAST)

function GetMyData(mySymbol,myField) as variant

GetMyData =application.worksheet

function.RTD("myDLLName.myClassModuleName,myMachineName,mySymbol,myField)

End function

7. The RTD function allows a maximum of 28 topics (strings which represent unique pieces of data in a real-time database) to be passed to the RTD server. Only one topic is required.
8. Excel does not automatically cache all data updates from the server. The RTD server must be designed to save all data updates until Excel is ready to accept them if dropped data is to be avoided.

9. No RTD servers come installed with Office XP. There are a couple of simple server examples available in the Microsoft knowledge base - see articles: **Q285339 - HOWTO: Create a RealTimeData Server for Excel 2002** and **Building a Real-Time Data Server in Excel 2002**. Another very informative article is: **Real-Time Data: Frequently Asked Questions**.

Summary

The addition of cell referencing to the real-time data mechanism is a welcome improvement. The use of DDE has been severely restricted without it. The fact that RTD is built into Excel and handles much of the communication overhead makes the creation of RTD servers relatively easy. On the other hand, switching from DDE to RTD as the method for bringing real-time data into spreadsheets would entail a major effort on the part developers to design, test, and phase in new servers. In addition, RTD is only available in Office XP. Many organizations have been slow to upgrade from Office 97 to Office 2000. It may be a while before XP becomes widespread enough to make the use of RTD a reality.